THE

CSAP

COMMON SECURITY ARCHITECTURE
for PRODUCTION

VERSION 1.3

PART 5A:
IMPLEMENTATION CONSIDERATIONS – STARTING OUT

Motion Picture Laboratories, Inc. August 1, 2023

Contents

1 Introduction

CSAP v1.3 is presented in six parts:

Part 1: Architecture Description the main architecture document.

Part 2: Interfaces describes the possible interfaces between the modules in a canonical form.

Part 3: Security Levels presents a metric-based approach to scaling security.

Part 4: Securing Software-Defined Workflow discusses how the security architecture can be applied to software-defined workflows that are managed using a service bus.

Part 5: Implementation Considerations is broken into sub-documents (5A, 5B, and 5C), which cover different aspects of CSAP implementation. This part is a new addition to CSAP version 1.2.

Part 6: Policy Description describes how policies and rules are defined. This part has not been published as of December 2022.

It is assumed that the reader is familiar with the previously published parts of CSAP, 1 to 4, and we do not reiterate the concepts described in those parts.

Changes from CSAP Part 1 v1.1

- The name of the *authorization policies* has been changed to *authorization rules*.
- The functions of the policy manager moved into the authorization service, the policy service in v1.0 is now the Authorization Rules Distribution Service (ARDS). In v1.0 this was called the policy engine. This does not change the functions necessary to create an authorization rule (formerly authorization policy), but consolidation simplifies this part of the architecture.
- Security initialization has been added.

Changes from CSAP Part 1 v1.2

- The functions of the Asset Protection Service have been merged into the authorization service. There is no change in function.
- The diagrammatic representation is now three services (authorization, authentication and the ARDS) as the part of the CSAP infrastructure.
- The CSAP supporting security functions Trust Inference and Continuous Trust Validation have been merged to reflect the direction of the market.
- This document has been expanded to include a description of the CSAP Zero Trust Foundation.

1.1 How to Use Part 5

In creating the CSAP architecture, the designers wanted the result to be implementable using existing technologies and with the least development possible. Part 5 provides a perspective on the way the CSAP designers ensured those goals were met. However, the descriptions of implementation

approaches may not represent the optimal approach and are not detailed enough to serve as implementation guides.

Part 5 is broken into sub-parts, each of which covers a particular aspect of implementation. Part 5 consists of three parts that cover some aspects of implementing CSAP, by no means all of them. We expect to update Part 5as developers gain experience in implementing CSAP.

- In Part 5A (this document) subtitled "Starting Out," Section 2 "CSAP Recap" and Section 3 "Implementation Basics" provide general guidance and sets the stage. Section 4 introduces the CSAP Zero-trust Foundation. Section 5 provides more detail on the CSAP Zero trust Foundation (ZTF) and section 6 discusses going from CSAP ZTF to CSAP level 100 and beyond.
- In Part 5B subtitled "CSAP Core," Section 2 "Identity and the Authentication Service" and Section 3 "Authorization and Authorization Rule Distribution Services" discuss implementation considerations for CSAP core security components. Section 4 "The User Experience" is a lesson in a way to create a good user experience.
- In Part 5C: subtitled "Approaches," Section 2 "The Network" covers ways in which networks may be used to support CSAP functions. Section 3 "Access Controls" discuss ways access to assets and resources can be controlled. Section 4 "End to End Security" looks at ways that the CSAP architecture can be used to facilitate end-to-end security on untrusted infrastructure terms.

Authentication is the security mechanism used to validate an entity's identity by a trusted authority. The entity might be a user, a service, a device, an application, etc.

Authorization is the security mechanism used by a trusted authority to determine whether an entity can perform an action.

An *Asset* is the broad term we use to mean any data and metadata that is part of the process of media creation including image data, sound data, and metadata. *This is the media definition of the word "asset," and not the definition used in cybersecurity where the word asset means any data, device, or other component (hardware or software) that supports information-related activities.*

The use of *Context* is the normal definition of that word, the setting, circumstances, or environment of an event. The MovieLabs Ontology for Media Creation has a specific and different meaning for *context,* which is not used in this document.

Policies are the abstract representation of what is to be authorized.[1]

Rules are the actionable representation of a policy.

A *Device* is a piece of infrastructure in the form of a physical or virtual system that serves as a platform for the execution of software.

[1] Or, in the very specific case of Global Security Policies, what is to be denied. Global Security Policies are the only place where a "deny" construct is needed since everywhere else, CSAP is deny by default.

Mutual authentication occurs when each entity that is part of forming a trust relationship can authenticate all the others. (As will be discussed later, in this context a user and their system may each be an entity.)

mTLS (mutual TLS) is a form of TLS with additional steps that authenticate the client to the server. In TLS, the server is authenticated to the client, but out-of-band authentication is required to authenticate the client to the server, e.g., using managed device services.

We use the terms for the structure of the organizations creating a creative work that are the common parlance of Hollywood, but they map directly to the terms used elsewhere.

The Studio is the entity that owns the rights to the creative work, is responsible for funding production and has a say (usually creative) in the production process. This is the same role as a commissioning broadcaster or a network (in the way that the term is used when referring to US linear broadcasters.) View the word "Studio" as a shorthand construct for anything that fits the definition.

Depending on the context, the term *The Production* is used to mean either:

- The entity responsible for carrying out production. This may be a production company, an organization set up to produce one creative work or a department or business unit that is part of the studio.

Or

- The complete process of producing the creative work.

Vendors are companies that provide services to the production. They may also be called production service providers. Examples are a VFX company, a transportation company, and a cloud infrastructure provider.

Please do not assume that our use of these terms means that CSAP is only for Hollywood studios or only for motion picture production. CSAP is for all types of media production including scripted and unscripted television.

1.2 Visual Language Security Icons

The shapes and icons used in the diagrams in this document are part of the MovieLabs Visual Language. Rather than add a key for the security icons to each diagram, we include it here.

Further information on the MovieLabs Visual Language can be found here at www.movielabs.com/production-technology/visual-language-for-media-creation/

1.3 Choice of Examples

In this document we present examples using commercial products and services. The choice of technology vendor, for example a cloud provider, is in no way an endorsement and does not imply that one product is better than another. In fact, where possible, we have steered away from examples where there is only one vendor.

We also discuss Google's BeyondCorp, and when we do, unless we say otherwise, we are talking about the zero-trust security solution Google implemented in their own offices and about which they have published a range of papers. These papers are cited frequently in the literature. Consider this to be an academic reference.

1.4 References

1.4.1 MovieLabs Publications

The Evolution of Production Workflows, MovieLabs, 2020

The Common Security Architecture for Production, Parts 1 to 4, MovieLabs.

1.4.2 Publications from Government Organizations

Zero Trust Architecture, NIST Special Publication 800-207, https://doi.org/10.6028/NIST.SP.800-207 .

2 CSAP Recap

Before we discuss implementation of CSAP, it is timely to review the components of CSAP and some of the concepts embodied in it. We recommend that anyone who has read previous versions of Part 1: Architecture review version 1.3, since the update to the architecture has changed the process of creating and distributing authorization rules.

2.1 Rules and Policies[2]

CSAP uses the terms policies and rules in this way:

- *Policies* are the abstract representation of what is to be authorized. They are what is requested by workflow management.
- *Rules* are the actionable representation of a policy. They are what is sent to the Policy Enforcement Points.

The CSAP designers anticipate that, in most cases, rules would be specific to the point of application.[3]

2.2 CSAP Components

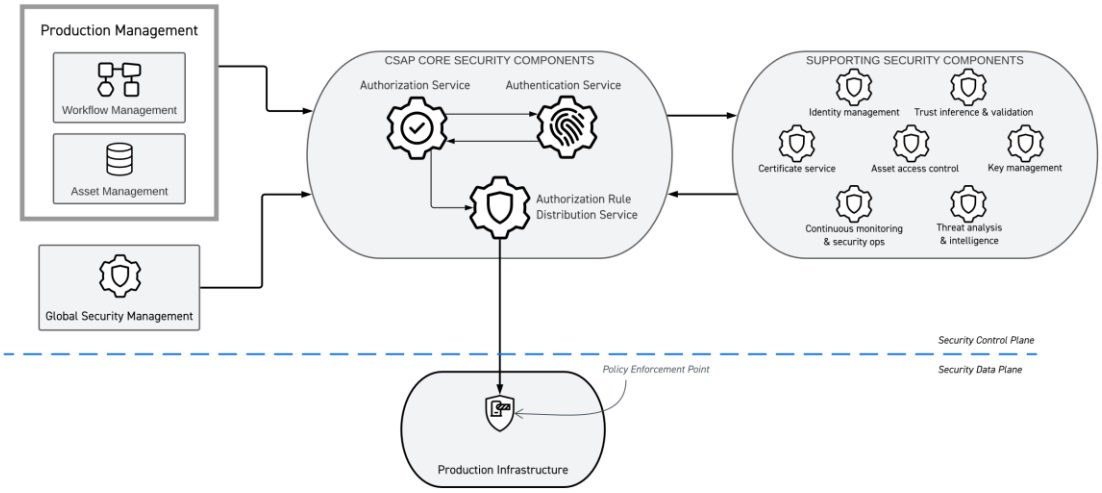

Figure 1 CSAP components

Four groupings of components are shown in:

[2] In CSAP v1.2 Part 1, the architecture was changed from a view where the Policy Enforcement Point interpreted the abstract representation to one that may be infrastructure specific.

[3] For example, cloud storage on a particular provider

Grouping	Explanation	Implementation examples
Core security components	Services providing: • Authorization service • Authentication service • Authorization rule distribution service (ARDS)	• Created specifically for CSAP • Commercial solutions customized for CSAP • Aggregation of solutions/services
Supporting security components	Services used by CSAP core components	• Commercial solutions configured for CSAP
Workflow management	External to CSAP, this is whatever is driving the workflow and is the source of authorization policy requests	• Software-defined workflow manager • Post-production house scheduling tool
Policy enforcement points	The point where CSAP interacts with the infrastructure, applications, workflow, etc.	• Service mesh proxy • Endpoint security • SaaS API • Storage ACLs • Asset encryption/decryption

Policy enforcement points are CSAP's data plane, and they interact with the workflow's control plane.

Note: this diagram does not address how the CSAP services are themselves protected. For example, the authorization service needs to only accept authorization policy requests from authenticated and authorized workflow management, and only accept global security policies from authenticated and authorized global security management.

2.3 Security Levels

Part 3 of the CSAP specifications describes three security levels, 100, 200 and 300, as a way of illustrating how CSAP security can be scaled according to the risk tolerance of a production.

The security levels are characterized by the availability of core and supporting security components, and the functionality/capabilities that are required to be available. Not all areas of a production will be at the same security level and not all technologies and services need to be capable of level 200 or level 300. For example:

- In a TV series, the security level for the season opener and the season finale might be set higher than for mid-season episodes.
- Certain scenes of a sensitive nature in a motion picture might warrant a higher level of security than has been set for the rest of the production.

Lastly, security can also be scaled, by which we mean the level of security can be turned up and down, by how the components are used. For example, if all access controls on the cloud storage are set according to group membership, it is likely that some members of the group will have more access privileges than their immediate task requires. CSAP does not set rules for how components are configured.

The tables below list the necessary and beneficial components. Changes from the previous level are bolded.

2.3.1 Level 100

Necessary Core Components	Beneficial Core Components

• Authentication service
• Authorization service
  ○ System wide capability for long lifetime authorization rules
  ○ Asset access controls
• Authorization rules distribution system (ARDS)
	• Authorization service
	  ○ Localized capability for medium and short lifetime authorization rules
	  ○ Local asset encryption capability
	  ○ End-to-end asset encryption capability

Necessary Supporting Components	Beneficial Supporting Components
• Identity management • Certificate service	• Trust inference • Continuous monitoring and security operations (CMSO) • Threat analysis and intelligence

2.3.2 Level 200

Necessary Core Components	Beneficial Core Components

• Authentication service
• Authorization service
  ○ System wide capability for long lifetime authorization rules
  ○ **Localized capability for medium** lifetime authorization rules
  ○ Asset access controls
  ○ **Local asset encryption capability**
• Authorization rules distribution system (ARDS)
	• Authorization service
	  ○ Localized capability for short lifetime authorization rules
	  ○ End-to-end asset encryption capability

Necessary Supporting Components	Beneficial Supporting Components
• Identity management • Certificate service	• Trust inference • Continuous monitoring and security operations (CMSO) CMSO • Threat analysis and intelligence

2.3.3 Level 300

Necessary Core Security Components	Beneficial Core Security Components

• Authentication service
• Authorization service
  ○ System wide capability for long lifetime authorization rules
  ○ **System wide capability for medium lifetime authorization rules**
  ○ **Localized capability for short lifetime authorization rules**
  ○ Asset access controls
  ○ Local asset encryption capability
  ○ **End-to-end asset encryption capability**
• Authorization rules distribution system (ARDS)	

Necessary Supporting Security Components	Beneficial Supporting Security Components
• **Identity management** • **Trust inference** • **Certificate service** • **Continuous monitoring and security operations (CMSO)** • **Threat analysis and intelligence**	

3 CSAP Implementation Basics

CSAP is a shift from a location-centric security model (e.g., perimeter) to a data and workflow-centric security model that is a better fit for media production in the cloud. It supports fine-grained security controls between participants, systems, assets, and workflows. This makes perfect sense in a world like the world of the MovieLabs 2030 Vision where either there is no centralized point of control or it is constantly moving.

Current security solutions are extrinsic, they are bolted onto or around the workflow systems. CSAP is security by design and, properly implemented, the security is intrinsic to the workflow. Apart from being a superior approach to security, this is one way that CSAP meets the 2030 Vision principle that security must not get in the way of the creative process.

In CSAP we, conceptually, make a distinction between authentication and authorization and the tenets of CSAP are:

1. Nothing can join any workflow unless it has been authenticated.
2. Nothing can join a specific workflow unless it has been authorized.

This means that everything must be authenticated and authorized before it can join a specific workflow. However, we describe it as we have done because authentication and authorization may not, in fact it is likely that they will not, originate from the same place. This isn't new to CSAP, it happens today. For example, a CG artist is hired and immediately given a task. When they try to log into their workstation, they discover that they are not yet in the corporate SSO system. Another example is that mid-way through a task an editor leaves the production and, as part of off boarding their account in the identity management system is locked.

Authentication means that something has been added to an identity management system used by the production's authentication service. The identity management system might be completely controlled by the production, it might be an SSO used across a larger organization, or it might be federated identity management with multiple owners. For example, someone working on a production may already be in the SSO identity management system of an organization (for example the studio) that the production uses. In that case, the organization would control who or what was in their identity management system.

On the other hand, authorization does come from the workflow management whether when a workflow is set up or as it proceeds. CSAP is workflow driven and does not require administrator intervention to authorize activity such access to assets. As we showed in our examples above, it is possible for the workflow management to authorize something to be part of a workflow that cannot be authenticated.

There is no "trust, but verify,"[4] no trusted networks (at least, not at the hardware level) where something is trusted because of the port it is connected to, and so on.

[4] The phrase means that someone is trusted to do something, but their compliance will be monitored/verified. See https://en.wikipedia.org/wiki/Trust,_but_verify

As we noted earlier, CSAP is zero-trust security applied to production. Thus, the journey from conventional or perimeter security to CSAP can be divided into two steps. The first step is the adoption of a zero-trust security model which is foundational to CSAP. The second step is to add the CSAP components and functions required by level 100.

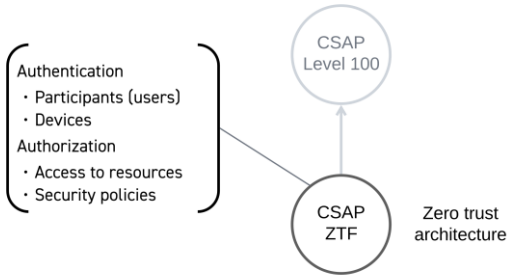

Figure 2 Zero-trust is foundational to CSAP

Zero-trust is primarily a change in security philosophy. It doesn't necessarily require new technology although it may require services that have not been deployed in a particular instance of a traditionally managed security environment.

CSAP is a Zero Trust architecture which means we must first have a common understanding of what trust means.

Mayer, Davis, and Schoorman (1995)[5] define trust as the willingness of a party to be vulnerable to the actions of another party based on the expectation that the other party will perform a particular action important to the trustor, irrespective of the ability to monitor or control that other party. This is an excellent definition for our purposes because it hints at the consequences of trusting something that is not trustworthy.

There are two factors in creating a trust relationship:

1. Determining whether an entity can be trusted
2. Determining where something claiming to be a trusted entity is indeed that entity and not an impostor

The first of these is outside of the scope of CSAP. It is a decision that is based on factors that vary from one organization to another, from one situation to another, and requires some form of risk assessment.

[5] Mayer, R.C., Davis, J.H., Schoorman, F.D., 1995. An integrative model of organizational trust. Academy of Management Review 20 (3), 709–734.

The second of those is the fundamental role of identity management. It is also important to avoid implicit[6] trust. For example, a trusted user's device should not be trusted just because that trusted user is using it.

Since trust is central to any zero-trust architecture including CSAP, robust identity management is a prerequisite.

3.1 Trust

If I say I trust you, I probably don't mean I trust you to do anything. I might trust a cardiac surgeon to perform heart surgery, but that doesn't mean I'm going to let them do brain surgery on me. Trust has boundaries.

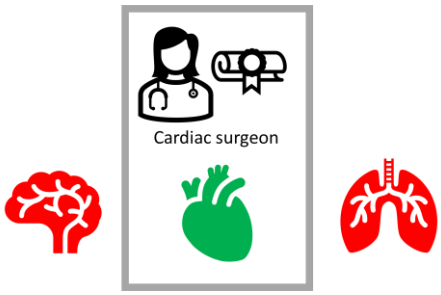

Figure 3 A trust boundary

A trust boundary[7] means that if something is authenticated it is trusted within that boundary but not (necessarily) outside of it. In our analogy, trusting heart surgeon to perform heart surgery but not pulmonary surgery or brain surgery defines the trust boundary. The heart is inside the trust boundary, the pulmonary system and the brain are outside.

A trust boundary is a combination of authentication and authorization. In that example, all the medical staff are authenticated prior to being authorized to do something. Furthermore, the surgeon can't just operate on any patient just because they have been authenticated. For a cardiac surgeon to perform surgery on you, someone (the patient for example) must allow (authorize) the surgeon to perform surgery. Even though the surgeon has been authenticated, you do not allow (authorize) the surgeon to perform pulmonary or brain surgery. Let's look at this as a workflow.[8]

1. A patient's doctor suspects a patient has a heart problem, so the doctor refers the patient to a cardiologist meaning the cardiologist is authorized to treat the patient.
2. The cardiologist determines the patient needs surgery and authorizes heart surgery.
3. The patient is admitted to hospital and prepared for surgery. Before surgery can commence, the surgeon and the anesthetist must agree it can commence (again, authorization).

[6] This is not the same as CSAP's Trust Inference where many factors such as location can be used to determine a trust level.
[7] Unrelated to a security perimeter.
[8] If this looks like a real medical workflow, it is a complete accident.

While this oversimplifies[9] the way the medical profession works, what we have is a workflow laid down by hospital policies, and the authorization to perform each step comes from that being the next step in the care process. This has many of the same properties as a media workflow.

Going back to authentication for a minute, the hospital effectively operates a zero-trust security model inasmuch they don't trust anyone to perform surgery just because they are in the hospital and wearing scrubs. The surgical staff must be authenticated one way or another.

3.2 Authorization

Media production workflows are complicated and dynamic and what needs to be authorized can be complicated and dynamic. There are many vendors, large and small, different departments, contract employees, studio employees, etc., all performing workflows that have a limited duration or change frequently. To authorize activity in this constant change is why CSAP has authorization rules that can be created on-demand as workflows progress. Simplistically, a CSAP authorization rule says who can do what using what resources to which assets and when it can be done.

CSAP's security is variable. Level 300 supports constantly changing authorization rules so that the principle of least privilege[10] can be applied as literally as required, including temporarily, but its policies can also be derived from the privileges that accompany user authentication in an IAM (identity and access management) system. CSAP does not require the security level to be consistent across the production.

So, yes, implementing CSAP authorization rules at the granularity possible in the architecture may not be easy right now, in part because it requires tight integration with currently nascent workflow management systems. However, there is no reason to start at the deep end. Authorization rules can be created and applied using existing access management systems. That supports the shift in the approach to security but draws on what is already in place.

All that having been said, separating out authentication and authorization is an essential part of the CSAP architecture, but it is conceptual, and it does not preclude authentication and authorization being implemented using an appropriate IAM system that handles both.

3.3 Trust inference

Trust inference is a level 300 requirement.

Note: We do not anticipate anyone will implement trust inference from scratch. This section describes how it can be implemented but the goal is to show considerations in evaluating trust inference services.

Trust inference is assessing whether behavior is normal. It is a property of a zero-trust architecture that it is assumed that the network is in a constant state of breach. Normal behavior of devices and users is learned from past behavior and trust inference is combining that with rules that say, for example, access

[9] An understatement!

[10] The security principle of least privilege means that the privileges given to a user should be only those that are required to conduct the immediate task and no more.

must never come from a particular set of locations. Conceptually, trust inference uses a trust rating based on a comparison of the context of the sign-in attempt (for example, IP address, device, time of day, etc.) and behavior learned about the user or device.

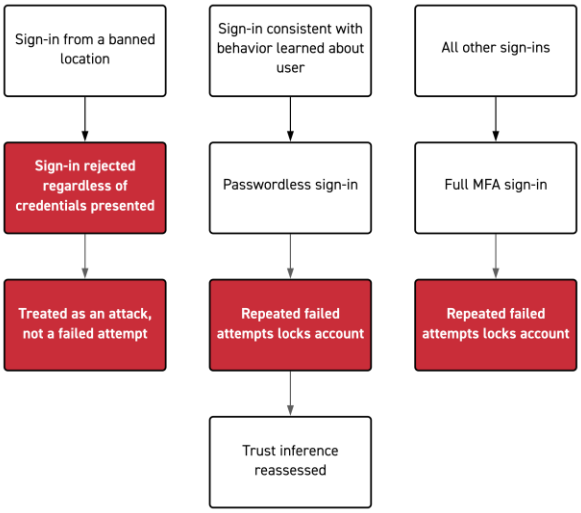

Figure 4 Trust inference at work

In our example, the circumstances of the sign-in attempt determine the level of credentials required but, to be very clear, the decision to use a reduced level of credential requirements is based on behavior learned about the user and is not solely a factor of where they are logging in from or the device they use.

Implementing trust inference requires a set of rules describing the circumstances when authentication attempts should always be rejected and a system that learns normal user and device behavior, and a processing engine that combines the two sets of data. Learning behavior is not confined to individual users or devices. It can also be role or group behavior. For example, someone performing the duties of an editor would have similar behavior to someone else performing the duties of an editor on the same production.

Continuous trust validation also requires information about a user's or device's current activity. See the next section for a description of how BeyondCorp trust inference works, particularly for devices.

3.3.1 BeyondCorp's Trust Tiers

By looking at how BeyondCorp[11] manages trust we get key pointers as to how trust inference and continuous trust evaluation can be implemented, the fact that trust does not need to evaluate to yes or no. The level of trust assigned can be a scale and, for example, some access is curtailed if the trust value is below a threshold.

[11] https://cloud.google.com/beyondcorp

The fundamental components of BeyondCorp are the Trust Inferer, device inventory services, access control engine, access policy, gateways, and resources.

Osborn, McWilliams, Beyer and Saltonstall [12] describe the BeyondCorp *Trust Inferer* which is a system that continuously analyzes and annotates device state. The system sets the maximum trust tier accessible by the device and assigns the VLAN to be used by the device on the corporate network. These data are recorded in the Device Inventory Service. Reevaluations are triggered either by state changes or by a failure to receive updates from a device.

BeyondCorp's trust tiers are assigned to each device by the Trust Inferer.

As a device is allowed to access more sensitive data, we require more frequent tests of user presence on the device, so the more we trust a given device, the shorter-lived its credentials. Therefore, limiting a device's trust tier to the minimum access requirement it needs means that its user is minimally interrupted. We may require installation of the latest operating system update within a few business days to retain a high trust tier, whereas devices on lower trust tiers may have slightly more relaxed timelines.

Trust inference for a device comes from two sets of data:

- Observed data generated from data collected such as the last time a security scan was performed on a device, the OS version and patch level, any installed software.
- Prescribed data which is manually assigned and might include the assigned owner of the device, users and groups allowed to access the device, DNS and DHCP assignments and explicit access to particular VLANs.

Data is analyzed from a variety of disparate sources to identify where data conflicts. There is no single or small number of systems that are regarded as the source of truth.

Trust tiers could also be used to assist the CSAP authorization service. As well as assigning a trust tier to each device, each resource is associated with a minimum trust tier required for access. To access a given resource, a device's trust tier assignment must be equal to or greater than the resource's minimum trust tier requirement.

[12] BeyondCorp: Design to Deployment at Google, Barclay Osborn, Justin McWilliams, Betsy Beyer Max Saltonstall, ;login:, :, vol. 41 (2016), pp. 28-34, https://www.usenix.org/system/files/login/articles/login_spring16_06_osborn.pdf

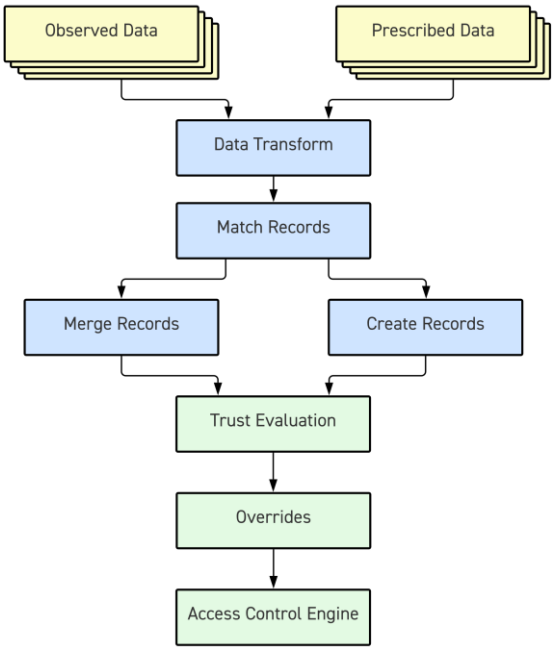

Figure 5 The BeyondCorp data processing pipeline (Source: Google)

Here the trust evaluation overrides the access control engine, and this is one way to implement not only initial trust inference but also CSAP's continuous trust evaluation.

Trust tiers could also be used to assist the CSAP authorization service. As well as assigning a trust tier to each device, each resource is associated with a minimum trust tier required for access. To access a given resource, a device's trust tier assignment must be equal to or greater than the resource's minimum trust tier requirement.

4 The CSAP Zero-Trust Foundation

If you are reading this document, you may be thinking to yourself that CSAP sounds like a good idea, but where to start?

The good news is that CSAP is not an architecture that stands apart from mainstream trends in cybersecurity. CSAP is aligned with a major shift in the cybersecurity landscape, the move to zero-trust security architectures. Simply put, CSAP is zero-trust architecture applied to the security of content production.

The CSAP Zero-Trust Foundation (ZTF) is a zero-trust implementation as might be used in any enterprise adopting zero-trust but with certain functionality that should be included.

Moving from traditional security to zero-trust is dependent on circumstances and not something this document will address. It all depends on where you start and, more importantly, there are many authoritative sources that can provide guidance and the technology for the transition to a zero-trust security architecture.

The CSAP ZTF is a zero-trust implementation as might be used in any enterprise but with particular characteristics necessary to fully implement CSAP. The requirement it places on the approach are not out of the ordinary and might be present in zero-trust implementations for other information technology system. CSAP ZTF it is not media production specific.

4.1 Purpose of the CSAP Zero-Trust Foundation

Zero-trust is not a well-defined term so if we say, “build CSAP on top of a zero-trust architecture” it isn’t helpful. In fact, there are many ways to define zero-trust, for example:

- Never trust, always verify. All network devices are untrusted until they have been authenticated, or;
- Zero Trust Architecture, NIST Special Publication 800-207. https://doi.org/10.6028/NIST.SP.800-207, or;
- How your current or potential security vendor defines it.

Obviously the first definition is useful in as much as you have an idea what it means but at a network level but that is not the best way to view zero-trust in a cloud environment. The NIST document is the best reference around but implementing it completely could, depending on your risk profile, result in something that is more complicated than is necessary for your needs. And the third one is too vague or might be a rebranding exercise and is one of the reasons we are defining the CSAP ZTF.

4.2 CSAP Zero-Trust Foundation Definition

CSAP ZTF is a zero-trust architecture implemented using the same off-the-shelf zero-trust solutions, for example those offered by leading cloud services providers, as any organization might use to implement zero-trust. Those solutions have a comprehensive array of features and a different selection might be made with different approaches.

Think of those solutions as a Tapas menu: usually you wouldn't eat absolutely everything on it, but CSAP ZTF are like the dishes you just must have!

Tapas menu. Source: The Square Orange, Keswick, UK

Figure 6 Allegorical view of zero-trust

Unlike perimeter security models, zero-trust architectures are deny-by-default and start with a very simple rule: everything must be authenticated before it can take part regardless of how it is connected. This leads us to the basic features required of a zero-trust implementation for it to be a CSAP ZTF:

1. It is universally deny-by-default.
    a. Nothing can take part in any workflow unless it has been appropriately authenticated. At minimum this applies to users, computer systems and services.
    b. Nothing can take part in a specific workflow unless it has been authorized to conduct the activity.
2. It has separate authentication and authorization services. Unlike perimeter security models, an authenticated user might present a token to a service, but authorization to do anything goes directly to the policy enforcement point associated with that service. (See the Figure 7.)
3. All authorizations are defined by security policies that are created and stored in an identifiable component of the system. This component becomes part of the CSAP Authorization Service.
4. The implementation assumes that the network is under the control of an intruder. The only exception would be if micro-segmentation is required for systems that have no options for intrinsic security, but the emphasis is on the word "micro."
5. All network traffic and system usage is continuously analyzed for abnormal activity.

Note that isn't a complete list of what is required in a zero-trust implementation. As we said, we're just making sure you include the recommended items on the Tapas menu.

4.3 Security Is Controlled by Policies

A zero-trust security implementation is driven by security policies – there is no trust, meaning there are no default authentication or authorization defaults. Building a CSAP ZTF means having an identifiable point or points that are the source of the security policies that say what can be authorized to do what. These policies are of the type "allow," meaning they permit activity – there is no need in zero-trust for policies that deny an activity since zero-trust is deny-by-default.[13] ("Deny" is implemented either by not authorizing something or, in for example the case of someone who has left the production, not authenticating.)

As you design your zero-trust implementation, the thread that holds it together is the policies.

Note that the management of security policies has to include a mechanism for changing or revoking those policies as well as a mechanism for creating and distributing them.

4.4 Authentication and Authorization Are Separate

In the other parts of CSAP and our blog posts we have described the two components of trust:

- Trustworthiness: determining if you can trust something
- Authentication: determining whether something is the trusted thing it claims to be

The first is a decision that you make using criteria that you create or take from other realms. The second is part of the security architecture and involves the presentation and validation of credentials.

This does not mean that authentication and authorization must be handled by different systems, although as CSAP functions are added doing so may prove more efficient, but the functions must be separable. For example, authenticating something should not provide an immutable set of authorizations as is the case with perimeter security when a user token from an identity and access management system includes access privileges.

[13] The only policies in CSAP that "deny" anything are the global security policies but they are not part of the CSAP Zero-Trust Foundation.

5 Implementing the CSAP Zero-trust Foundation

Regardless of what your security system is today, put it aside for the moment and formulate a plan to implement a zero-trust security solution that meets the needs of CSAP ZTF.

Rather than looking at this as problem of getting from A (your current security implementation) to B (zero-trust security), we suggest the best place to start is to determine how to implement zero-trust for your systems. Yes, you are at A and, yes, you need to get to B, but starting with what zero-trust looks like and how you implement it gives you a clear goal.

There is a wealth of literature and solution providers out there to help you with implementing zero-trust and we have a short reading list in the appendix. To reemphasize the point, CSAP ZTF is not a media specific zero-trust implementation, it's a zero-trust solution that might be implemented in any organization. CSAP ZTF has required features that not all zero-trust solutions might have.

Two factors are relevant to your existing security solution:

1. What is my zero-trust deployment process?
2. What can be kept and re-used?

On the first point, for example, if part of your infrastructure is on-premises, adequately secured, and does not need to interact with systems (external or internal) built on the principles of the MovieLabs 2030 Vision then you might decide to put that further out on your deployment schedule.

On the second point, re-use isn't just about (say) keeping your identity management system. It can go deeper, for example can I keep the same access controls on assets such as access control lists (ACLs) or role-based access control (RBAC)? Or is changing attribute-based access control (ABAC) or relationship-based access control (ReBAC) a better proposition? But it is important to know how you plan to deploy zero-trust before considering what you can re-use.

One more example that bridges the two points: if you are using microsegmentation for a small group of systems, and it is truly secure, then perhaps you keep it and focus on deploying a policy enforcement point at the point where the microsegment is accessed.

In the rest of this section, we will walk through an approach to implementing the CSAP Zero-trust Foundation. At the end of each subsection, we'll state what is needed for the CSAP ZTF.

5.1 Collapsing the Protect Surface

John Kindervag, often credited with defining zero-trust, defines "protect surface" as the thing you are trying to protect, it is the attacker's target, and it is where you put your protection measures. The protect surface is as close as possible around the thing that is protected.

Kindervag uses the Secret Service's method for protecting the US President as an example.

Rather than relying on a security perimeter around the neighborhood the presidential motorcade is driving through, the Secret Service's protect surface is reduced to the president's vehicle. The protect

surface is guarded by the agents walking alongside the vehicle working with the agents inside the vehicle and in conjunction with the agents in the following vehicle.

Kindervag calls the uniformed agents and police officers standing along the street "security theater."[14] Their role is to protect against the low hanging fruit, for examples individuals in the crowd who charge toward the president's vehicle and to intimidate anyone planning an attack. But the protect surface is around the president's vehicle.

In applying this analogy to a system, the first step is to define the protect surfaces. Protect surfaces may be around each server in your system, as is the case with the services in a mesh network, or if that is impractical for a system then network microsegmentation might be used. In this latter strategy, the operative word is segmentation, and everything on that segment must have a reason for being there. For example, it's unlikely that data ingest systems need to be in the same network segment as rendering nodes because they are at opposite ends of a VFX workflow.

The CSAP zero-trust foundation needs the protect surfaces to be as small as possible.

The protect surfaces relate directly to the "blast radius", the area of impact, of a security breach.

5.2 Map Workflows

Whether you are starting with an existing infrastructure protected by a traditional perimeter security approach or you are building new workflows on a cloud infrastructure, you need to start by mapping your workflows so that you understand who will be doing what tasks and with what assets and infrastructure.

One of the advantages we have securing production over someone securing a corporate network is that our workflows are known and, at least to some extent, documented. You know how your dailies workflow works; you know how your VFX rendering workflow works. In the corporate environment, workflows are generally opaque[15] to the IT department and require exploration before zero-trust can be implemented.

The CSAP zero-trust foundation needs to be implemented for specific workflows and the connections between them (at whatever level of granularity you choose to interpret that).

5.3 Create Policies

Once you have your protect surfaces defined, meaning you know exactly what you are protecting and they are as small as possible, and you know your workflows, you are ready to architect your system and deploy it.

[14] The term security theater was coined by computer security expert Bruce Schneier in his book Beyond Fear. He has applied the term to the TSA security measures introduced at airports following 9/11.

[15] The IT department provides email services but they don't necessarily know how they are used other than to send messages to other people. For example, emails that recap decisions made in a meeting, sending email to yourself as a way of making notes, or filing emails in folders as a way of tracking different contract negotiations.

Each policy must authorize as little as possible; to reduce complexity and increase manageability it is better to have many policies authorizing similar things than have a single multi-part policy that covers everything. Each policy should be only as complicated as is necessary to authorize a particular part of the workflow at a particular point in time. For example, one policy might authorize access to assets by authorizing access to the storage location, and another policy authorizes access to a SaaS service.

It is likely that every policy will have components that are specific to a particular infrastructure, for example, a policy authorizing access to assets on one cloud provider's infrastructure may be different from a policy that authorizes the same activity of a different cloud provider's infrastructure. In CSAP we define two classes of polices:

- Authorization Policies are an abstract expression of what is authorized.
- Authorization Rules are Authorization Policies translated to the specific needs of a particular infrastructure.

It isn't necessary to make that distinction when implementing the CSAP ZTF, however doing so will probably reduce the complexity of processing the policies at the policy enforcement point.

The CSAP zero-trust foundation requires that activities, for example accessing assets, are controlled using policies maintaining a separation between authentication and authorization.

5.4 Policy Enforcement Points

A zero-trust architecture has policy enforcement points where a decision is made on whether something is authorized by a policy. In CSAP, we refer to policies in this context as Authorization Policies, and one of the parameters in those policies is the identity of the user (or more generally, the Participant) that is authorized to conduct the activity. The policy enforcement point accepts the user's identity token and uses that in combination with the authorization policy to determine if the activity is authorized.

This differs from a traditional approach where the user's token includes access claims (meaning what they are authorized to do). In CSAP, the authorization policy is not a property stored in the user's record in the identity management system.

The two approaches can be seen in this diagram with the conventional approach on the left and the zero-trust approach on the right.

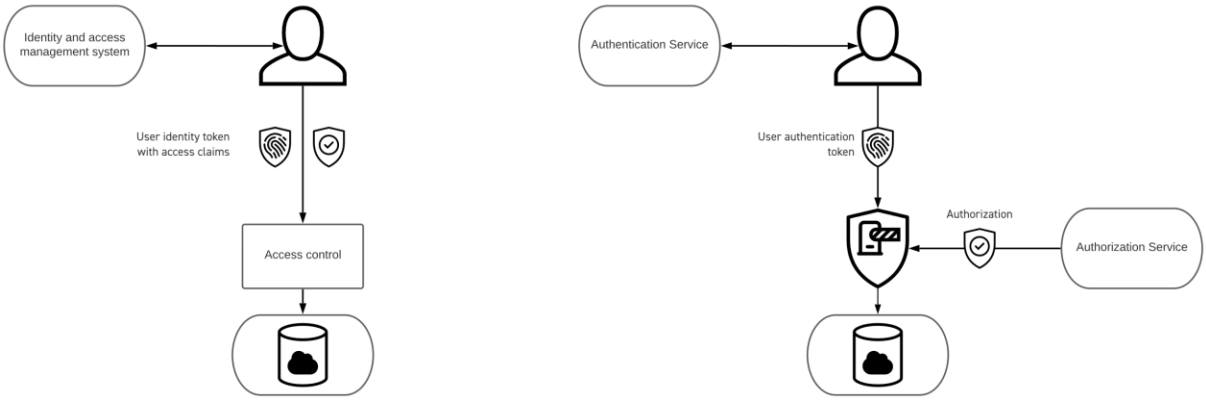

Figure 7 Conventional vs. zero-trust authorization

This is the manifestation of CSAP zero-trust foundation requirements that activities, for example accessing assets, are controlled using policies maintaining a separation between authentication and authorization.

In both cases, authentication and access privileges/authorization are required and processed before the user can access assets. The difference is that in the conventional approach (left) access privileges are part of the user token which was created when the user logged it whereas in the zero-trust case (right), authorization is managed by policies which can be fine grained, of limited lifetime and generally better managed.

In CSAP, we refer to NIST's policy enforcement points and policy decision points collectively as policy enforcement points. All the functions are still there. We anticipated that there will be many ways that the policy enforcement point can be implemented. In some cases, the policy enforcement point is implemented using native security components of the infrastructure (for example, access controls in storage) in which case what we call a policy enforcement point may be setting the infrastructure access controls.

In the CSAP ZTF, policies determine what is authorized rather than rights, permissions, privileges or claims maintained as part of identity management.

5.5 Analytics and Monitoring

Analytics is a level 300 requirement but should be present in any zero-trust system.

Zero-trust requires pervasive instrumentation.

The starting point in zero-trust networking is the assumption that the network has been breached. The starting point in perimeter security is that network breaches can be prevented.

In perimeter security, activity analytics are looking for abnormal behavior. It looks for activity that indicates that the network has been breached. Those analytics might be using fingerprints associated with malicious activity. If a fingerprint of malicious activity is the movement of large amounts of data,

that's not going work in a production environment where moving large amounts of data is normal. Furthermore, malicious data movement can be buried in the noise floor of legitimate activity.

Analytics in a zero-trust network needs to do more: it needs to learn what is normal. That has two purposes: activity that the analytics determines is normal might not need the same amount of scrutiny; and identifying normal behavior is a key part of trust inference.

This approach does present challenges. Creating a picture of normal behavior while assuming that the network is in a constant state of breach means avoiding polluting the picture of normal behavior with unauthorized behavior assumed to be present on the network. While a thesis on how to solve that dilemma is, fortunately, beyond the scope of this document, we posit that the answer lies in the fact that production workflows are known.

CrowdStrike states:

> *Enforcement of Zero Trust policies rely on real-time visibility into 100's of user and application identity attributes such as:*
>
> - *User identity and type of credential (human, programmatic)*
> - *Credential privileges on each device*
> - *Normal connections for the credential and device (behavior patterns)*
> - *Endpoint hardware type and function*
> - *Geo location*
> - *Firmware versions*
> - *Authentication protocol and risk*
> - *Operating system versions and patch levels*
> - *Applications installed on endpoint*
> - *Security or incident detections including suspicious activity and attack recognition*

Zero-trust does not assume that the security "just works". It requires continuous analytics to ensure that it is working the way it is supposed to work.

6 CSAP ZTF to CSAP

From the CSAP Zero-Trust Foundation, CSAP functionality is added on top to enable implementations to achieve CSAP level 100. Thus, the path might be:

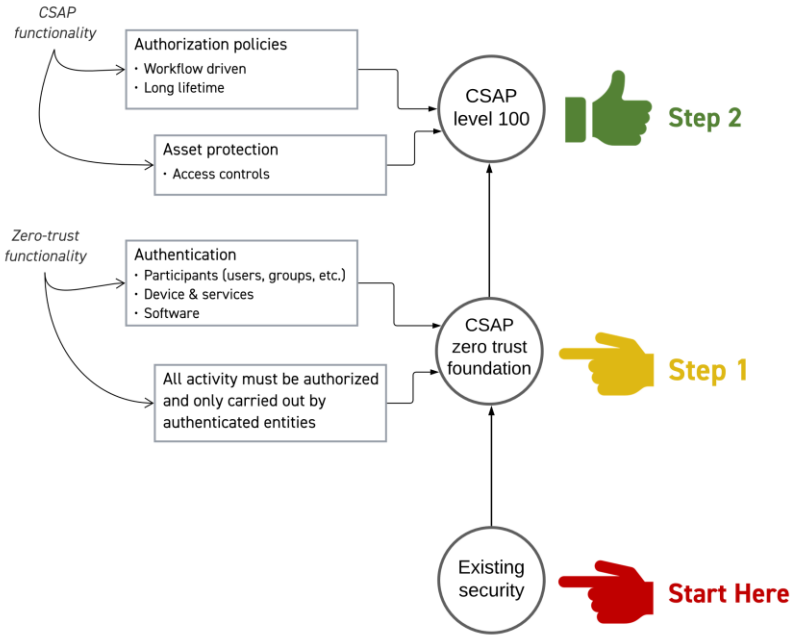

Figure 8 Zero to CSAP zero-trust

6.1 Getting to Level 100

The step from the CSAP zero-trust foundation to level 100 requires implementing the necessary functionality which of course means that the associated core and supporting components must be available.

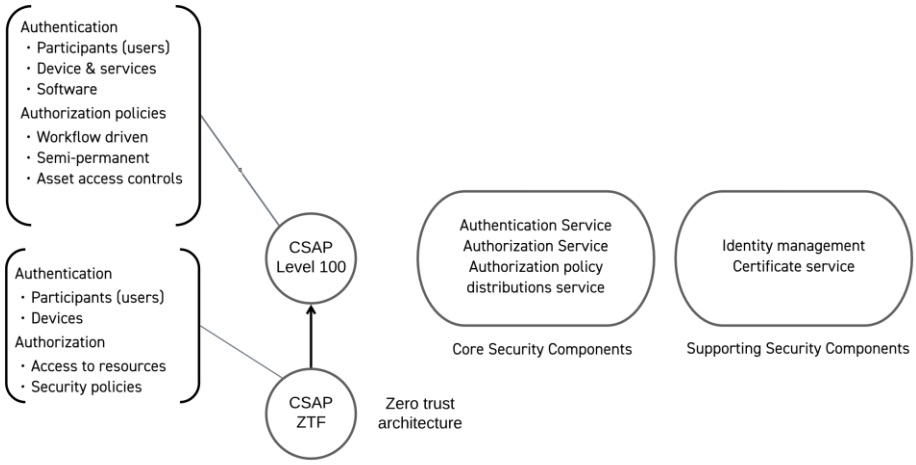

Figure 6-9 Transitioning from CSAP zero-trust foundation to CSAP level 100

The core components may be specific services implemented for CSAP or they may be, to a greater or lesser extent, embodied in existing services.

The authorization service, ARDS, and possibly the authentication service are new services that can be run at the production or vendor levels. The service providers create PEPs and can define the actions/templates used by their PEP when those templates are used to create service specific authentication rules. It is the authorization policy request that provides the data to the authorization service that it used to transform those templates into authorization rules.

Here is where we see CSAP's central enhancement of zero-trust: security, and in particular authorization, is controlled by the production workflow.

6.2 Getting to Levels 200 and 300

Whether security needs to meet the CSAP level 200 or 300 requirements is a matter for the production to decide. We anticipate that, initially at least, only parts of a production and certain services and technologies will need to be at level 200 or 300.

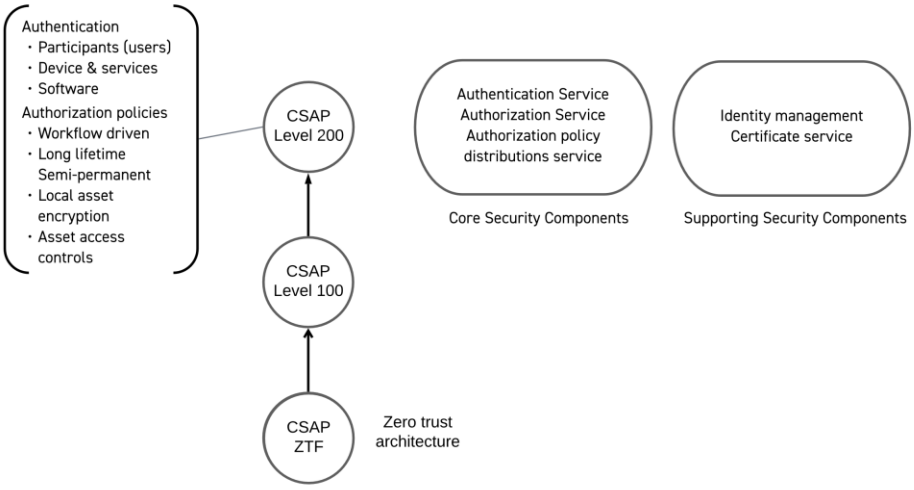

Figure 6-10 Progression from level 100 to level 200

As we move up the CSAP levels we primarily see that there is more granularity in the authorization rules, a shift from assets protected by access controls to assets encrypted individually in a way that is not a property of the storage infrastructure, and more capabilities are end-to-end leading to the end-to-end asset encryption in level 300.

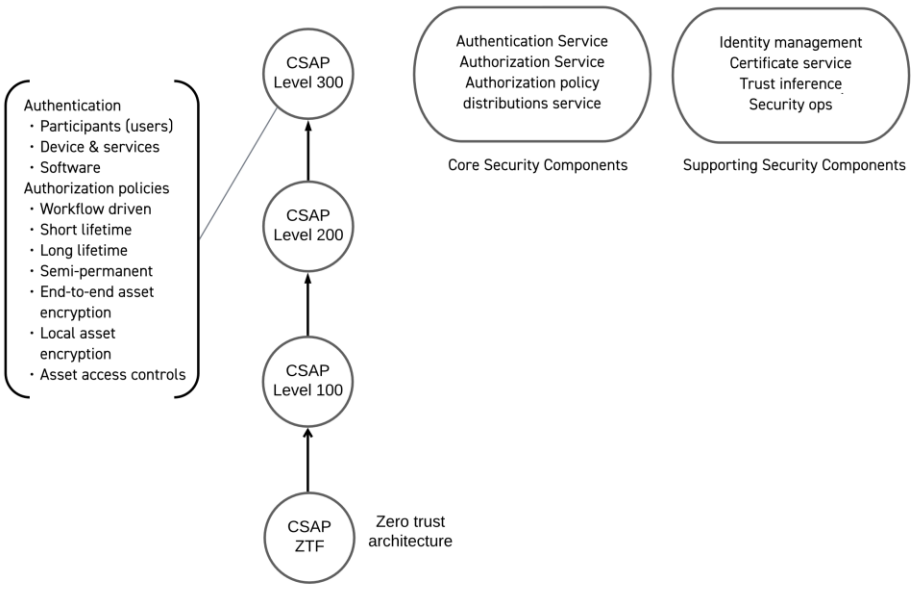

Figure 6-11 Progression from level 200 to level 300

Level 300 also requires the use of additional supporting security components. Some of these serve other security components. For example, continuous monitoring serves trust inference while trust inference itself is required to improve the user experience that may otherwise be negatively affected by the more granular security of level 300.

7 Conclusion

The CSAP zero-trust foundation has characteristics are common to other zero-trust system although not necessarily present in all “zero-trust” products.

Appendix A Suggested Reading

If you wish to understand more about zero-trust architectures, we have homework for you.

Zero Trust Networks: Building Secure Systems in Untrusted Networks by Evan Gilman and Doug Barth, O'Reilly, ISBN: 1491962194.

Zero Trust Architecture by Scott W. Rose, Oliver Borchert, Stuart Mitchell, Sean Connelly, NIST Special Publication 800-207, https://doi.org/10.6028/NIST.SP.800-207

Zero Trust Security: An Enterprise Guide by Jason Garbis and Jerry W. Chapman, Apress, ISBN 148426701X

Project Zero Trust: A Story about a Strategy for Aligning Security and the Business, 1st Edition by George Finney (Author), John Kindervag (Foreword) ISBN 1119884845

BeyondCorp: A New Approach to Enterprise Security by Rory Ward and Betsy Beyer, Google Research. https://research.google/pubs/pub43231/

AWS [insert link], Google Cloud Platform [insert link] and Azure have useful documentation on using their security services to assist you in building zero-trust security into your cloud platform, however we believe that having a basic understanding of CSAP will help you deciding how to use those services to create the CSAP ZTF.

Here is a short (and incomplete) list of resources that can help with transitioning to a zero-trust security model:

Transitioning to modern access architecture with Zero Trust, Microsoft, https://www.microsoft.com/en-us/insidetrack/transitioning-to-modern-access-architecture-with-zero-trust

A unified and proven Zero Trust system with BeyondCorp and BeyondProd, Google Cloud, https://cloud.google.com/blog/products/identity-security/applying-zero-trust-to-user-access-and-production-services

Zero Trust on AWS. Advancing your security model with a Zero Trust approach, AWS, https://aws.amazon.com/security/zero-trust/ (this is set of resources including videos)

Zero Trust. A revolutionary approach to Cyber or just another buzz word?, Deloitte, https://www2.deloitte.com/content/dam/Deloitte/de/Documents/risk/deloitte-cyber-zero-trust.pdf

5 Steps to Zero Trust. A simple guide to deploying Zero Trust networks, Palo Alto Networks. https://start.paloaltonetworks.com/5-steps-to-zero-trust.html (registration required)

Cisco Zero Trust Architecture Guide, Cisco, https://www.cisco.com/c/en/us/solutions/collateral/enterprise/design-zone-security/zt-arch-guide.html

IBM Security Zero Trust Acceleration Services. Accelerate zero trust adoption, IBM Security, https://www.ibm.com/downloads/cas/DBG68MKM

Getting Started with Zero Trust Access Management: Trust Begins with Secure Identity, Okta, https://www.okta.com/resources/whitepaper-getting-started-with-zero-trust-access-management-wbs/ (registration required)

MovieLabs does not endorse any of these solutions and some may present options that are not consistent with the CSAP ZTF.